

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Please amend Claims 1, 7, 10-12, and 15 as follows.

1 (currently amended): A computer implemented method for efficient customization of a VHDL technology library, the method comprising:

storing in a memory a tpd_super_rise generic declaration and a tpd_super_fall generic declaration for every VHDL gate model in a VHDL technology library;

initializing other generic variables, each derived solely from at least one of the tpd_super_rise generic declaration and the tpd_super_fall generic declaration, corresponding to every VHDL gate model in the VHDL technology library to an equation representing a correlation policy; and

storing in a memory an updated VHDL technology library including

the tpd_super_rise generic declaration and the tpd_super_fall generic declaration for every VHDL gate model, and

the initialized other generic variables.

2 (original): The method of claim 1, wherein the correlation policy comprises:

collecting all generic variables in a VHDL standard delay file;

selecting a generic variable; and

extracting all delay values for the selected generic variable.

3-6 (cancelled).

7 (currently amended): A system comprising:

a processor/controller; and

a memory for storing a VHDL technology library and a VHDL technology library modifier, the memory communicatively coupled to the processor/controller, for:

inserting a `tpd_super_rise` generic declaration and a `tpd_super_fall` generic declaration for at least one VHDL gate model in the VHDL technology library[[,]] :

initializing other generic variables, each derived solely from at least one of the `tpd_super_rise` generic declaration and the `tpd_super_fall` generic declaration, in every VHDL gate model in the VHDL technology library to an equation representing a correlation policy[[,]] ;
and

storing an updated VHDL technology library including the `tpd_super_rise` generic declaration and the `tpd_super_fall` generic declaration for the at least one VHDL gate model, and including the initialized other generic variables.

8 (original): The system of claim 7, further comprising:

the memory for storing a VHDL correlation file and a VHDL standard delay file; and

a program memory, communicatively coupled to the processor/controller and the memory, for storing a VHDL simulator, and for binding correlated delay constants in a 3-dimensional variable data array structure to a VHDL technology library.

9 (original): The system of claim 8, wherein the VHDL correlation file comprises a VHDL package embedded with correlation delay data.

10 (currently amended): A computer program product for updating a VHDL technology library for efficient customization of chip gate delays, the computer program product comprising:

a storage medium readable by a processing circuit and storing instructions for execution by the a processing circuit for performing a method comprising:

storing a tpd_super_rise generic declaration and a tpd_super_fall generic declaration for every VHDL gate model in a VHDL technology library;

initializing other generic variables, each derived solely from at least one of the tpd_super_rise generic declaration and the tpd_super_fall generic declaration, corresponding to every VHDL gate model in the VHDL technology library to an equation representing a correlation policy; and

storing an updated VHDL technology library including:

the tpd_super_rise generic declaration;

the tpd_super_fall generic declaration for every VHDL gate model; and

the initialized other generic variables

11 (currently amended): The computer program product ~~readable medium~~ of claim 10, wherein the correlation policy comprises:

collecting all generic variables in a VHDL standard delay file;

selecting a generic variable; and

extracting all delay values for the selected generic variable.

12 (currently amended): A computer readable medium comprising instructions for:

binding correlated delay constants in a 3-dimensional variable data array structure to a VHDL technology library using a VHDL package embedded with correlation delay data, wherein the VHDL technology library includes:

one or more VHDL gate models;

a tpd_super_rise generic declaration and a tpd_super_fall generic declaration for every VHDL gate model in the VHDL technology library; and

one or more other generic variables, each derived solely from at least one of the tpd_super_rise generic declaration and the tpd_super_fall generic declaration, corresponding to every VHDL gate model in the VHDL technology library initialized to an equation representing a correlation policy.

13 (original): The computer readable medium of claim 12 wherein the 3-dimensional variable data array structure comprises:

a z-axis representing a set of common blocks for each logical topology of a VHDL logic gate;

an x-axis representing a delay name for the gate topology; and

a y-axis representing an actual delay value.

14 (original): The computer readable medium of claim 13, wherein the z-axis of the data structure represents a generic delay name common to a plurality of logic gates.

15 (currently amended): A computer readable medium comprising instructions for:

using a `tpd_super_rise` generic declaration and a `tpd_super_fall` generic declaration, each generic declaration comprising at least one pointer, for every VHDL gate model in a VHDL technology library to index into a 3-dimensional variable data array structure comprising delay values, wherein the VHDL technology library includes:

one or more VHDL gate models;

a `tpd_super_rise` generic declaration and a `tpd_super_fall` generic declaration for every VHDL gate model in the VHDL technology library; and

one or more other generic variables, each derived solely from at least one of the `tpd_super_rise` generic declaration and the `tpd_super_fall` generic declaration, corresponding to every VHDL gate model in the VHDL technology library initialized to an equation representing a correlation policy; and

resolving the pointers when VHDL modules are linked together.